

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Сыров Игорь Анатольевич
Должность: Директор
Дата подписания: 22.08.2023 10:52:04
Уникальный программный ключ:
b683afe664d7e9f64175886cf9626a196149ad36

СТЕРЛИТАМАКСКИЙ ФИЛИАЛ
ФЕДЕРАЛЬНОГО ГОСУДАРСТВЕННОГО БЮДЖЕТНОГО ОБРАЗОВАТЕЛЬНОГО
УЧРЕЖДЕНИЯ ВЫСШЕГО ОБРАЗОВАНИЯ
«УФИМСКИЙ УНИВЕРСИТЕТ НАУКИ И ТЕХНОЛОГИЙ»

Факультет Математики и информационных технологий
Кафедра Прикладной информатики и программирования

Аннотация рабочей программы дисциплины (модуля)

дисциплина *Технологии и методы программирования*

Блок Б1, базовая часть, Б1.Б.21

цикл дисциплины и его часть (базовая, вариативная, дисциплина по выбору)

Направление

10.03.01

Информационная безопасность

код

наименование направления

Программа

Безопасность компьютерных систем (по отрасли или в сфере профессиональной деятельности)

Форма обучения

Очная

Для поступивших на обучение в
2020 г.

1. Перечень планируемых результатов обучения по дисциплине (модулю)

1.1. Перечень планируемых результатов освоения образовательной программы

Выпускник, освоивший программу высшего образования, в рамках изучаемой дисциплины, должен обладать компетенциями, соответствующими видам профессиональной деятельности, на которые ориентирована программа:

Способностью понимать значение информации в развитии современного общества, применять информационные технологии для поиска и обработки информации (ОПК-4)

1.2. Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с планируемыми результатами освоения образовательной программы

Формируемая компетенция (с указанием кода)	Этапы формирования компетенции	Планируемые результаты обучения по дисциплине (модулю)
Способностью понимать значение информации в развитии современного общества, применять информационные технологии для поиска и обработки информации (ОПК-4)	1 этап: Знания	Обучающийся должен знать: современные компьютерные технологии и программные средства, используемые при разработке ПО; основные принципы создания программ на изучаемом языке программирования; синтаксис и семантику изучаемого языка программирования;
	2 этап: Умения	Обучающийся должен уметь: применять на практике полученные знания и использовать доступные программные средства для создания программ; транслировать текстовую задачу в программный код; искать альтернативные пути решения, если для решения поставленной задачи не хватает инструментов языка;
	3 этап: Владения (навыки / опыт деятельности)	Обучающийся должен владеть: основами работы в известных программных оболочках для решения прикладных задач путем написания программ; основными понятиями и принципами, используемыми в программировании; навыками написания программ;

2. Место дисциплины (модуля) в структуре образовательной программы

Дисциплина реализуется в рамках базовой части. Для освоения дисциплины необходимы знания, полученные в рамках школьного курса информатики, а также курса «Алгоритмы и

языки программирования» .

Дисциплина «Технологии и методы программирования» занимает важное место среди изучаемых дисциплин. В процессе работы студенты должны на основе изученных тем и рассмотренных примеров приобрести практические навыки и умения в конструировании программ для ЭВМ, предназначенных для решения различных прикладных задач, в том числе связанных с информационными процессами и системами.

Дисциплина изучается на 1 курсе в 2 семестре

3. Объем дисциплины (модуля) в зачетных единицах с указанием количества академических или астрономических часов, выделенных на контактную работу обучающихся с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающихся

Общая трудоемкость (объем) дисциплины составляет 4 зач. ед., 144 акад. ч.

Объем дисциплины	Всего часов
	Очная форма обучения
Общая трудоемкость дисциплины	144
Учебных часов на контактную работу с преподавателем:	
лекций	12
практических (семинарских)	18
лабораторных	18
другие формы контактной работы (ФКР)	1,2
Учебных часов на контроль (включая часы подготовки):	34,8
экзамен	
Учебных часов на самостоятельную работу обучающихся (СР)	60

Формы контроля	Семестры
экзамен	2

4. Содержание дисциплины (модуля), структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий

4.1. Разделы дисциплины и трудоемкость по видам учебных занятий (в академических часах)

№ п/п	Наименование раздела / темы дисциплины	Виды учебных занятий, включая самостоятельную работу обучающихся и трудоемкость (в часах)			
		Контактная работа с преподавателем			СР
		Лек	Пр/Сем	Лаб	
1	Объектно-ориентированное программирование на C++.	12	18	18	60
1.1	Классы в C++.	2	4	4	12
1.2	Разработка структуры классов.	2	2	4	8
1.3	Дружественные функции и классы.	2	4	0	12
1.4	Иерархия классов.	2	2	4	8
1.5	Усложнение структуры классов.	2	4	2	10
1.6	Перегрузка операций.	2	2	4	10
	Итого	12	18	18	60

4.2. Содержание дисциплины, структурированное по разделам (темам)

Курс лекционных занятий

№	Наименование раздела / темы дисциплины	Содержание
1	Объектно-ориентированное программирование на C++.	
1.1	Классы в C++.	Основные понятия. Классы. Конструкторы и деструкторы. Область действия и доступ к членам класса.
1.2	Разработка структуры классов.	Статические поля. Конструктор копирования. Указатель this. Передача значения по ссылке.
1.3	Дружественные функции и классы.	Организация взаимодействия нескольких объектов разных классов посредством дружественных функций. Взаимодействие классов на примере описания структуры «Клиент – Банк».
1.4	Иерархия классов.	Инкапсуляция. Вложенные классы в C++. Наследование и полиморфизм. Множественное наследование. Виртуальные методы.
1.5	Усложнение структуры классов.	Абстрактные классы. Виртуальные базовые классы. Шаблоны классов. Нетипизированные аргументы. Частичная специализация шаблона
1.6	Перегрузка операций.	Механизм перегрузки операций. Перегрузка унарных и бинарных операций. Перегрузка операции вызова функции и операции индексирования. Перегрузка операций «поместить в поток» и «взять из потока».

Курс лабораторных занятий

№	Наименование раздела / темы дисциплины	Содержание
1	Объектно-ориентированное программирование на C++.	
1.1	Классы в C++.	Лабораторная работа №1. Объекты и классы. Введение. Разработка классов для описания математических и геометрических объектов. Функции-утилиты. Конструкторы с параметрами по умолчанию.
1.2	Разработка структуры классов.	Лабораторная работа №2. Производные классы. Наследование. Создание и усложнение классов на основе ранее описанных более простых классов. Практическое применение механизма наследования при описании объектов более сложной структуры.
1.4	Иерархия классов.	Лабораторная работа №3. Полиморфизм. Виртуальные функции. Полиморфное поведение функций при наследовании. Раннее и позднее связывание. Необходимость описания функции как виртуальной.
1.5	Усложнение структуры классов.	Лабораторная работа №4. Виртуальные функции. Абстрактные классы. Практика описания и использования абстрактных классов и чисто виртуальных функций.
1.6	Перегрузка операций.	Лабораторная работа №5. Перегрузка операций. Практическое использование механизма перегрузки операций при работе с экземплярами объектов различных классов.

Курс практических/семинарских занятий

№	Наименование раздела / темы дисциплины	Содержание
1	Объектно-ориентированное программирование на C++.	
1.1	Классы в C++.	Классы для описания простых математических и геометрических объектов. Инкапсуляция. Примеры разработки классов для различных областей использования.
1.2	Разработка структуры классов.	Организация взаимодействия нескольких объектов разных классов посредством дружественных функций. Взаимодействие классов на отдельных примерах.
1.3	Дружественные функции и классы.	Механизм наследования. Множественное наследование. Конструкторы и деструкторы базовых и производных классов. Раннее и позднее связывание, виртуальные функции
1.4	Иерархия классов.	Назначение и использование абстрактных классов. Абстрактные классы как основа более сложных структурных объектов. Невозможность создания экземпляров объектов абстрактных классов. Чисто виртуальные функции.
1.5	Усложнение структуры классов.	Механизм перегрузки операций. Перегрузка унарных и бинарных операций. Перегрузка операции вызова функции и операции индексирования. Перегрузка операций «поместить в поток» и «взять из потока».
1.6	Перегрузка операций.	Практическое применение и обобщение приемов и методов разработки классов для описания математических и других (физических, социальных, и т.д.) объектов.